



## Bayesian Design of Experiments Using Approximate Coordinate Exchange

Antony M. Overstall & David C. Woods

To cite this article: Antony M. Overstall & David C. Woods (2017) Bayesian Design of Experiments Using Approximate Coordinate Exchange, *Technometrics*, 59:4, 458-470, DOI: [10.1080/00401706.2016.1251495](https://doi.org/10.1080/00401706.2016.1251495)

To link to this article: <https://doi.org/10.1080/00401706.2016.1251495>



© 2017 The Author(s). Published with license by Taylor & Francis© Antony M. Overstall and David C. Woods



[View supplementary material](#)



Accepted author version posted online: 27 Oct 2016.  
Published online: 27 Apr 2017.



[Submit your article to this journal](#)



Article views: 1353



[View Crossmark data](#)



Citing articles: 11 [View citing articles](#)

# Bayesian Design of Experiments Using Approximate Coordinate Exchange

Antony M. Overstall and David C. Woods

Southampton Statistical Sciences Research Institute, University of Southampton, Southampton, United Kingdom

## ABSTRACT

The construction of decision-theoretical Bayesian designs for realistically complex nonlinear models is computationally challenging, as it requires the optimization of analytically intractable expected utility functions over high-dimensional design spaces. We provide the most general solution to date for this problem through a novel approximate coordinate exchange algorithm. This methodology uses a Gaussian process emulator to approximate the expected utility as a function of a single design coordinate in a series of conditional optimization steps. It has flexibility to address problems for any choice of utility function and for a wide range of statistical models with different numbers of variables, numbers of runs and randomization restrictions. In contrast to existing approaches to Bayesian design, the method can find multi-variable designs in large numbers of runs without resorting to asymptotic approximations to the posterior distribution or expected utility. The methodology is demonstrated on a variety of challenging examples of practical importance, including design for pharmacokinetic models and design for mixed models with discrete data. For many of these models, Bayesian designs are not currently available. Comparisons are made to results from the literature, and to designs obtained from asymptotic approximations. Supplementary materials for this article are available online.

## ARTICLE HISTORY

Received January 2016

Accepted September 2016

## KEYWORDS

Computer experiments;  
Gaussian process emulator;  
High-dimensional design;  
Smoothing

## 1. Introduction


Bayesian design of experiments is a natural paradigm for many problems arising in the physical sciences and engineering, particularly those concerning the estimation of nonlinear models where design performance, as measured by classical optimality criteria, is dependent on the a priori unknown values of the model parameters. A decision-theoretic approach, reviewed by Chaloner and Verdinelli (1995), determines an optimal allocation of experimental resources via maximization of the expected utility

$$U(\delta) = \int \int_{\Psi, \mathcal{Y}} u(\delta, \psi, \mathbf{y}) \pi(\mathbf{y}, \psi | \delta) d\mathbf{y} d\psi. \quad (1)$$

Here, the utility  $u(\delta, \psi, \mathbf{y})$  quantifies the experimenter's gain from using design  $\delta \in \mathcal{D}$  to obtain data  $\mathbf{y} \in \mathcal{Y}$  assuming model parameter values  $\psi \in \Psi$ , with the statistical model defined through the joint density function  $\pi(\mathbf{y}, \psi | \delta) = \pi(\mathbf{y} | \psi, \delta) \pi(\psi)$ . As an example, assume the  $i$ th response is modeled as  $y_i = g(x_i; \theta) + \varepsilon_i$  with the  $x_i$  defining values taken by a controllable variable,  $\theta$  being a vector of unknown parameters defining the mean response, and observation error  $\varepsilon_i \sim N(0, \sigma^2)$  ( $i = 1, \dots, n$ ). Then  $\psi^T = (\theta^T, \sigma^2)$ ,  $\delta = (x_1, \dots, x_n)^T$ , and likelihood  $\pi(\mathbf{y} | \psi, \delta)$  is a multivariate normal density function. The utility function  $u(\delta, \psi, \mathbf{y})$  will typically be a function of some posterior quantities of  $\psi$  (see Section 3.1).

Selection of a *fully Bayesian* optimal design  $\delta^* = \arg \max_{\delta \in \mathcal{D}} U(\delta)$  has traditionally been challenging for all but the most straightforward utility functions and models due to the high-dimensional and, typically, analytically intractable integrals in (1). Some recent progress has been made using simulation-based methodologies for low-dimensional problems, that is, small numbers of controllable variables and/or small numbers of design points, see Ryan et al. (2016) and references therein. There are, however, no methods available for decision-theoretic Bayesian optimal, or near-optimal, multi-variable design for nonlinear models. The methodology in this article fills this important gap, and is demonstrated on generic problems of practical importance including pharmacokinetic studies and experiments that produce discrete data. Previous attempts to obtain fully Bayesian optimal designs for these types of experiment have been extremely limited.

In a landmark article for low-dimensional design problems, Müller and Parmigiani (1996) proposed selection of a design by maximizing a surrogate function found by approximating  $U(\delta)$  for a small number,  $m$ , of designs using simulation, and then smoothing the resulting values,  $\tilde{U}(\delta_1), \dots, \tilde{U}(\delta_m)$ . See also Jones et al. (2016) and Weaver et al. (2016). In essence, these approaches perform a computer experiment to construct a statistical *emulator* for the approximation  $\tilde{U}(\delta)$ , a research area where there has been huge activity in recent years (see, e.g., Dean et al. 2015, sec. V). For an experiment with  $n$  runs and  $v$  variables,  $\delta$  has  $nv$  elements.

**CONTACT** Antony M. Overstall  [A.M.Overstall@southampton.ac.uk](mailto:A.M.Overstall@southampton.ac.uk)  Southampton Statistical Sciences Research Institute, University of Southampton, Southampton SO17 1BJ, United Kingdom.

 Supplemental data for this article can be accessed on the publisher's website at <http://www.tandfonline.com/r/TECH>.

© 2017 Antony M. Overstall and David C. Woods. Published with license by Taylor & Francis.

This is an Open Access article distributed under the terms of the Creative Commons Attribution License (<http://creativecommons.org/licenses/by/4.0/>), which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Therefore, application of this approach to design for multi-variable models suffers from a curse of dimensionality, requiring (i) the construction of emulators in very high dimensions; (ii) large, for example, space-filling, designs composed of selections of points from an  $nv$ -dimensional space, leading to (iii) a prohibitive number of evaluations of  $\tilde{U}(\delta)$ , particularly if  $\tilde{U}(\delta)$  is computationally expensive.

Our approach overcomes these problems by building a series of one-dimensional emulators for the approximated expected utility. We emulate  $\tilde{U}(\delta) = \tilde{U}(\delta_i | \delta_{(i)})$  as a function of only the  $i$ th “coordinate” (or element)  $\delta_i$  conditional on  $\delta_{(i)} = (\delta_1, \dots, \delta_{i-1}, \delta_{i+1}, \delta_{nv})^T$ , the values of all coordinates excluding the  $i$ th ( $i = 1, \dots, nv$ ). When these emulators are combined with a continuous version of the coordinate exchange algorithm (Meyer and Nachtsheim 1995), an effective and computationally efficient design selection methodology results. Conditional, coordinate-wise, optimization is key to overcoming the curse of dimensionality described above.

Until relatively recently, the usual approach to Bayesian design was to use a normal distribution as an asymptotic approximation to the posterior distribution of  $\psi$  (e.g., Chaloner and Larntz 1989). For standard utility functions (see Section 3.1.2), use of such a *pseudo-Bayesian* approach leads to the integrand in (1) no longer depending on the data  $y$ . The resulting integral, with respect to  $\psi$ , typically has much lower dimension and can be approximated using efficient deterministic quadrature rules (Gotwalt, Jones, and Steinberg 2009). However, the appropriateness of such approximations for small experiments is open to question.

For high-dimensional design, an alternative to the use of a normal approximation was suggested by Ryan et al. (2014). These authors combined the simulation-based approach of Müller (1999) (see also Müller, Sanso, and De Iorio 2004; Amzal et al. 2006) with a dimension-reduction scheme to find designs for single-variable nonlinear models ( $v = 1$ ) variables and a large number of runs. Designs were restricted to those formed from a sampling scheme defined via two parameters, for example, the initial design point and a spacing parameter. An optimal design in this subclass then consists of the best choices of these two parameters, a substantially easier optimization problem to solve.

In contrast to either applying an asymptotic approximation or restricting attention to a subset of the design space, both of which may result in the selection of inefficient designs with respect to the exact expected utility, we attempt to find optimal or efficient designs for the original problem across the whole design space via an approximate optimization scheme. These three different approaches are compared in Section 3.

The remainder of the article is organized as follows. In Section 2, we describe approximate coordinate exchange for finding decision-theoretic Bayesian designs, including the use of Monte Carlo integration and Gaussian process emulators to approximate the expected utility. The methods are applied to a range of challenging and practically relevant examples in Section 3 including models for which Bayesian design has previously been computationally infeasible. We summarize the advantages of our approach in Section 4 and highlight some ongoing work.

## 2. Approximate Coordinate Exchange (ACE)

We first establish some notation. Suppose that a design consists of  $n$  runs or points, each of which determines the settings of  $v$  controllable variables and results in a single observation of the response variable. Let  $\mathbf{D}$  denote the  $n \times v$  design matrix with  $k$ th row  $\mathbf{d}_k$  specifying the settings of the  $v$  factors in the  $k$ th run ( $k = 1, \dots, n$ ). Let  $q = nv$ , then the design may be represented as a  $q$ -vector  $\delta = \text{vec}(\mathbf{D}) \in \mathcal{D} \subset \mathbb{R}^q$ , where  $\text{vec}(\cdot)$  denotes vectorization via stacking the columns of a matrix and  $\mathcal{D}$  is the  $q$ -dimensional design space.

The proposed algorithm for decision-theoretic Bayesian design has two phases. Phase I applies a novel coordinate exchange algorithm where, for each coordinate, maximization of  $U(\delta)$  is replaced by maximization of a surrogate function  $\hat{U}(\delta)$ . Phase I tends to produce clusters of design points that are very similar in the values of the controllable variables. Such clustering is common in heuristic design search (see also Gotwalt, Jones, and Steinberg 2009). Hence, in Phase II, we check if the points in each cluster can be consolidated into a replicated design point using a point exchange algorithm (Atkinson, Donev, and Tobias 2007, chap. 12). Replication of points is common in optimal design for parametric models and a key principle of design of experiments (Wu and Hamada 2009, chap. 1). In Phase II, the candidate set is the design found from Phase I. The two phases form an approximate coordinate and point exchange algorithm which, for brevity, we call the ACE algorithm.

In Section 2.1, we define the ACE algorithm. For Steps 2a–2c of the algorithm, we assume the availability of (i) a Monte Carlo approximation of the expected utility,

$$\tilde{U}(\delta_i | \delta_{(i)}) = \tilde{U}(\delta) = \sum_{l=1}^B u(\delta, \mathbf{y}_l, \psi_l) / B, \quad (2)$$

with  $\{\mathbf{y}_l, \psi_l\}_{l=1}^B$  a random sample from the joint distribution with density  $\pi(\mathbf{y}, \psi | \delta)$ ; (ii) *coordinate-designs*  $\xi_i = \{\delta_i^1, \dots, \delta_i^m\} \in \mathcal{D}_i$  at which we evaluate  $\tilde{U}(\delta_i | \delta_{(i)})$ , where  $\mathcal{D}_i \subset \mathbb{R}$  is the domain for the  $i$ th coordinate; and (iii) a suitable one-dimensional emulator,  $\hat{U}(\delta_i | \delta_{(i)})$ , for  $\tilde{U}(\delta_i | \delta_{(i)})$ . Further details are given in Section 2.2, with examples in Section 2.4.

ACE is designed to solve a stochastic optimization problem, as only approximations to the expected utility are available formed as linear combinations of realizations of the random variable  $u(\delta, \mathbf{y}, \psi)$ . As such, proposed changes to the design in Steps 2d and 4e of the algorithm are accepted with probability derived from a Bayesian test of the difference in the means of Monte Carlo approximations to the expected utility for the current and proposed designs. Further details are given in Section 2.3.

The R package *acebayes* (Overstall et al. 2017) provides an implementation of ACE and is available on CRAN.

### 2.1 The ACE Algorithm

1. Choose an initial design  $\delta^0$  and set the current design  $\delta^C = \delta^0$ .

### Phase I: Coordinate exchange.

Phase I finds designs through a sequence of one-dimensional computer experiments. For each coordinate of the design, a computer experiment is performed where (i) all other design coordinates are taken as fixed; (ii) the numerical approximation (2) is evaluated for a small number of values of the single coordinate (Steps 2a and 2b below), and (iii) a Gaussian process surrogate is constructed and maximized to suggest a new value of the coordinate (Steps 2c and 2d). As discussed above, this suggested coordinate is accepted with probability obtained from a Bayesian test of equality of the Monte Carlo approximations of the expected utility for the original design and the design with the new coordinate, reflecting the stochastic nature of the Monte Carlo approximation. An optimal design is then found via repetition of this sequence of computer experiments (Step 3). A simple example of the application of Phase I is given in Section 2.4.

2. For  $i = 1, \dots, q$ ,
  - (a) Select an  $m$  point coordinate-design  $\xi_i = \{\delta_i^1, \dots, \delta_i^m\} \in \mathcal{D}_i$ .
  - (b) Evaluate  $\tilde{U}(\delta_i^1 | \delta_{(i)}^C), \dots, \tilde{U}(\delta_i^m | \delta_{(i)}^C)$ .
  - (c) Construct  $\hat{U}(\delta_i | \delta_{(i)}^C)$  by fitting a statistical model to  $\{\delta_i^j, \tilde{U}(\delta_i^j)\}_{j=1}^m$ .
  - (d) With probability  $p_i^\dagger$ , set  $\delta_i^C = \delta_i^\dagger = \arg \max_{\delta_i \in \mathcal{D}_i} \hat{U}(\delta_i | \delta_{(i)}^C)$ , where

$$p_i^\dagger = 1 - T_{2B-2} \left( -\frac{B\tilde{U}(\delta_i^C) - B\tilde{U}(\delta_i^\dagger)}{\sqrt{2B\hat{v}_i}} \right), \quad (3)$$

$T_{2B-2}$  is the probability distribution function for a  $t$  distribution with  $2B - 2$  degrees of freedom,  $\delta_i^C = (\delta_i^C, \dots, \delta_{i-1}^C, \delta_i^\dagger, \delta_{i+1}^C, \dots, \delta_q^C)^T$ , and

$$\hat{v}_i = \frac{\sum_{l=1}^B [u(\delta_i^C, \mathbf{y}_l^C, \boldsymbol{\psi}_l^C) - \tilde{U}(\delta_i^C)]^2 + \sum_{l=1}^B [u(\delta_i^\dagger, \mathbf{y}_l^C, \boldsymbol{\psi}_l^C) - \tilde{U}(\delta_i^\dagger)]^2}{2B - 2},$$

for  $\{\mathbf{y}_l^\dagger, \boldsymbol{\psi}_l^\dagger\}_{l=1}^B$  and  $\{\mathbf{y}_l^C, \boldsymbol{\psi}_l^C\}_{l=1}^B$  independent random samples from  $\pi(\mathbf{y}, \boldsymbol{\psi} | \delta_i^C)$  and  $\pi(\mathbf{y}, \boldsymbol{\psi} | \delta_i^\dagger)$ , respectively.

3. Repeat Step 2  $N_I$  times.

### Phase II: Point exchange.

Phase II consolidates clusters of similar points in a design arising from Phase I. A point exchange algorithm is employed with a candidate list formed from the points in the Phase I design. First, the design point is found whose replication maximizes the approximation to the expected utility. A replicate of this point is then added to the design (Steps 4a and 4b). Second, from this new  $(n + 1)$ -point design, the point is found whose deletion leads to the  $n$ -point design with highest approximated expected utility (Steps 4c and 4d). Finally, the new design with these two points swapped is accepted with probability once again arising from a Bayesian test for equality of the Monte Carlo approximations (Step 4e). These steps are replicated to find an optimal design (Step 5).

4.

- (a) For  $k = 1, \dots, n$ , let  $\delta_k^{(1)} = \text{vec}(\mathbf{D}_k^{(1)})$ , where

$$\mathbf{D}_k^{(1)} = [(\mathbf{D}^C)^T (\mathbf{d}_k^C)^T]^T,$$

and  $\mathbf{d}_k^C$  is the  $k$ th row of  $\mathbf{D}^C$ , the design matrix for  $\delta^C$ .

- (b) Let  $k^\dagger = \arg \max_k \tilde{U}(\delta_k^{(1)})$  and set  $\mathbf{D}^{(2)} = \mathbf{D}_{k^\dagger}^{(1)}$ .
- (c) For  $h = 1, \dots, n + 1$ , let  $\delta_h^{(3)} = \text{vec}(\mathbf{D}_h^{(3)})$ , where

$$\mathbf{D}_h^{(3)} = \left[ (\mathbf{d}_1^{(2)})^T \dots (\mathbf{d}_{h-1}^{(2)})^T (\mathbf{d}_{h+1}^{(2)})^T \dots (\mathbf{d}_{n+1}^{(2)})^T \right]^T,$$

and  $\mathbf{d}_h^{(2)}$  is the  $h$ th row of  $\mathbf{D}^{(2)}$ .

- (d) Let  $h^\dagger = \arg \max_h \tilde{U}(\delta_h^{(3)})$
- (e) With probability  $p_{II}^\dagger$ , set  $\delta^C = \delta_{h^\dagger}^{(3)}$ , where

$$p_{II}^\dagger = 1 - T_{2B-2} \left( -\frac{B\tilde{U}(\delta_{h^\dagger}^{(3)}) - B\tilde{U}(\delta^C)}{\sqrt{2B\hat{v}_{II}}} \right) \quad (4)$$

and

$$\hat{v}_{II} = \frac{\sum_{l=1}^B [u(\delta_{h^\dagger}^{(3)}, \mathbf{y}_l^{(3)}, \boldsymbol{\psi}_l^{(3)}) - \tilde{U}(\delta_{h^\dagger}^{(3)})]^2 + \sum_{l=1}^B [u(\delta^C, \mathbf{y}_l^C, \boldsymbol{\psi}_l^C) - \tilde{U}(\delta^C)]^2}{2B - 2},$$

with  $\{\mathbf{y}_l^{(3)}, \boldsymbol{\psi}_l^{(3)}\}_{l=1}^B$  a random sample from  $\pi(\mathbf{y}, \boldsymbol{\psi} | \delta_{h^\dagger}^{(3)})$ .

5. Repeat Step 4  $N_{II}$  times.
6. Return  $\delta^* = \delta^C$ .

The decision on when to terminate a run of the algorithm, that is, choice of  $N_I$  and  $N_{II}$ , is complicated by the stochastic nature of the approximation to the expected utility. For the examples in Section 3,  $N_I = 20$  and  $N_{II} = 100$  are sufficient to achieve approximate convergence. Here, convergence is assessed graphically from trace plots of  $\tilde{U}(\delta^C)$  against iteration number; see Section 3.2 for examples of such plots.

To avoid local optima, the algorithm is run  $M$  times (in embarrassingly parallel fashion) with each run starting from a different, randomly chosen, initial design  $\delta_0$  (a random Latin hypercube design, unless otherwise stated). The selected design,  $\delta^*$ , is the design having the highest average approximate expected utility, averaged across  $C$  sets of Monte Carlo simulations. In this article,  $M = C = 20$  was used, unless otherwise stated.

## 2.2 Emulation via Computer Experiments (Steps 2a–2c)

In Phase I of the algorithm, a sequence of one-dimensional emulators is constructed for  $\tilde{U}(\delta_i | \delta_{(i)}^C)$ ,  $i = 1, \dots, q$  (Step 2c). A variety of smoothing or interpolation techniques could be applied to construct each emulator. Müller and Parmigiani (1996) used local polynomial regression to emulate low-dimensional design utilities. We adopt a Gaussian Process (GP) regression model (see, e.g., Rasmussen and Williams 2006), which is widely used for computer experiments, and use the posterior predictive mean as an emulator. Let

$$\hat{\mu}_i = \sum_{j=1}^m \tilde{U}(\delta_i^j | \delta_{(i)}^C) / m,$$



$$\hat{\sigma}_i^2 = \sum_{j=1}^m (\tilde{U}(\delta_i^j | \delta_{(i)}^C) - \hat{\mu}_i)^2 / (m-1),$$

and  $z(\delta_i) = (U(\delta_i | \delta_{(i)}^C) - \hat{\mu}_i) / \hat{\sigma}_i$  for any  $\delta_i \in \mathcal{D}_i$ . The GP model assumes that any vector  $\mathbf{z}(\zeta) = [z(\delta^1), \dots, z(\delta^{m_0})]^T$ , for  $\zeta = \{\delta^1, \dots, \delta^{m_0}\} \in \mathcal{D}_i^{m_0}$  and integer  $m_0$ , has joint distribution  $N(\mathbf{0}_{m_0}, \mathbf{A}(\zeta))$ , with  $\mathbf{0}_{m_0}$  the  $m_0$  zero-vector and  $\mathbf{A}(\zeta)$  an  $m_0 \times m_0$  covariance matrix. Hence, the posterior predictive mean of  $\tilde{U}(\delta | \delta_{(i)})$  at an arbitrary  $\delta \in \mathcal{D}_i$  can be derived using standard results on the conditional distribution of normal random variables and used as an emulator:

$$\begin{aligned} \hat{U}(\delta | \delta_{(i)}^C) &= \hat{\mu}_i + \hat{\sigma}_i E[z(\delta) | \mathbf{z}(\xi_i)] \\ &= \hat{\mu}_i + \hat{\sigma}_i \mathbf{a}(\delta, \xi_i)^T \mathbf{A}(\xi_i)^{-1} \mathbf{z}(\xi_i). \end{aligned}$$

Under the common assumption of a squared exponential correlation function,  $\mathbf{A}(\xi_i)$  and  $\mathbf{a}(\delta, \xi_i)$  have entries

$$\begin{aligned} \mathbf{A}(\xi_i)_{st} &= \exp \left\{ -\rho (\delta_i^s - \delta_i^t)^2 \right\} + \eta I(r=s), \\ \mathbf{a}(\delta, \xi_i)_s &= \exp \left\{ -\rho (\delta_i^s - \delta)^2 \right\}, \end{aligned}$$

for  $s, t = 1, \dots, m$ , where  $I(E)$  is the indicator function for the event  $E$ , and  $\rho, \eta > 0$  are unknown parameters. The inclusion of nugget  $\eta$  ensures the emulator will smooth, rather than interpolate, the Monte Carlo approximations of the expected utility. To limit computational complexity, at each iteration we find maximum likelihood estimates of  $\rho$  and  $\eta$  via Fisher scoring (see, e.g., Pawitan 2001, pp. 174–177). In contrast, a fully Bayesian approach would require application of a Markov chain Monte Carlo algorithm to construct each emulator, substantially increasing the computational cost of the algorithm.

At each iteration of Step 2a, a coordinate-design  $\xi_i = (\delta_i^1, \dots, \delta_i^m)$  must be chosen at which to evaluate  $\tilde{U}(\delta_i | \delta_{(i)})$ . We use a space-filling design, specifically a randomly selected one-dimensional Latin hypercube design (see, e.g., Santner, Williams, and Notz 2003, chap. 5), constructed by dividing  $\mathcal{D}_i$  into  $m$  equally sized sub-intervals, and then generating a point at random from each interval. We set  $m = 20$ , unless otherwise stated. This choice of  $m$  is conservative relative to the rule of thumb (Loeppky, Sacks, and Welch 2009) of setting  $m$  equal to 10 times the number of input dimensions (suggesting  $m = 10$  in our case). We have, however, found it works well in practice for a variety of different types of examples, giving accurate emulators and not being overly computationally demanding.

### 2.3 Adjusting a Design Coordinate (Step 2d) or Point (Step 4e)

To make a change to the  $i$ th coordinate in Step 2d, we first find  $\delta_i^\dagger$ , the value of the coordinate that maximizes the emulator. We find the maximum by evaluating  $\hat{U}(\delta | \delta_{(i)}^C)$  for 10,000 uniformly generated points in  $\mathcal{D}_i$ . This discretization of the problem has proved both more reliable than continuous optimization and sufficiently computationally efficient.

Choice of  $\delta_i^\dagger$  is subject to both Monte Carlo error, from the evaluation of  $\tilde{U}(\delta_i | \delta_{(i)})$ , and emulator error from the estimation of  $\hat{U}(\delta_i | \delta_{(i)})$  resulting, for example, from an inappropriate choice of correlation function or errors in estimating  $\rho$

and  $\eta$ . It is clearly impossible to use usual residual diagnostics (Bastos and O'Hagan 2009) to check emulator adequacy at each iteration of the algorithm. Instead, emulator error is eliminated from the decision to adjust a design coordinate by performing additional Monte Carlo integration to calculate the probability  $p_I^\dagger$  in (3). This quantity is the posterior probability that  $E[u(\boldsymbol{\psi}, \mathbf{y}, \delta^{C\dagger})] > E[u(\boldsymbol{\psi}, \mathbf{y}, \delta^C)]$  under noninformative prior distributions and using Monte Carlo samples  $\{\boldsymbol{\psi}_l^C, \mathbf{y}_l^C\}_{l=1}^B$  and  $\{\boldsymbol{\psi}_l^\dagger, \mathbf{y}_l^\dagger\}_{l=1}^B$ , assuming both  $u(\boldsymbol{\psi}^\dagger, \mathbf{y}^\dagger, \delta^{C\dagger})$  and  $u(\boldsymbol{\psi}^C, \mathbf{y}^C, \delta^C)$  are normally distributed with equal variances. See also Wang and Zhang (2006) for use of a classical hypothesis test in a simulated annealing algorithm. If this normality assumption were severely violated, a more sophisticated test procedure could be adopted at greater computational cost.

A similar test is performed at Step 4e in Phase 2 of the algorithm to calculate  $p_{II}^\dagger$  in (4). We demonstrate the effect of Step 2d in the next section.

### 2.4 Illustrative Example

In this section, we illustrate the ACE methodology, in particular the combination of Steps 2c and 2d in selecting and accepting a proposed change to the design. To enable assessment of the algorithm, we consider the analytically tractable problem of finding a one-point optimal design for the single-variable Poisson model  $y|\beta \sim \text{Poisson}(e^{\beta x})$ . There is a single design coordinate,  $\delta = x \in [-1, 1]$ , and hence our notation is simplified by replacing  $\delta$  by  $x$  in this example. A priori, we assume  $\beta \sim N(0.5, 1)$  and adopt the utility function that leads to pseudo-Bayesian  $D$ -optimality (Section 3.1.2), given by

$$\begin{aligned} u(\beta, y, x) &= \log \mathcal{I}(\beta; x) \\ &= 2 \log |x| + \beta x, \end{aligned}$$

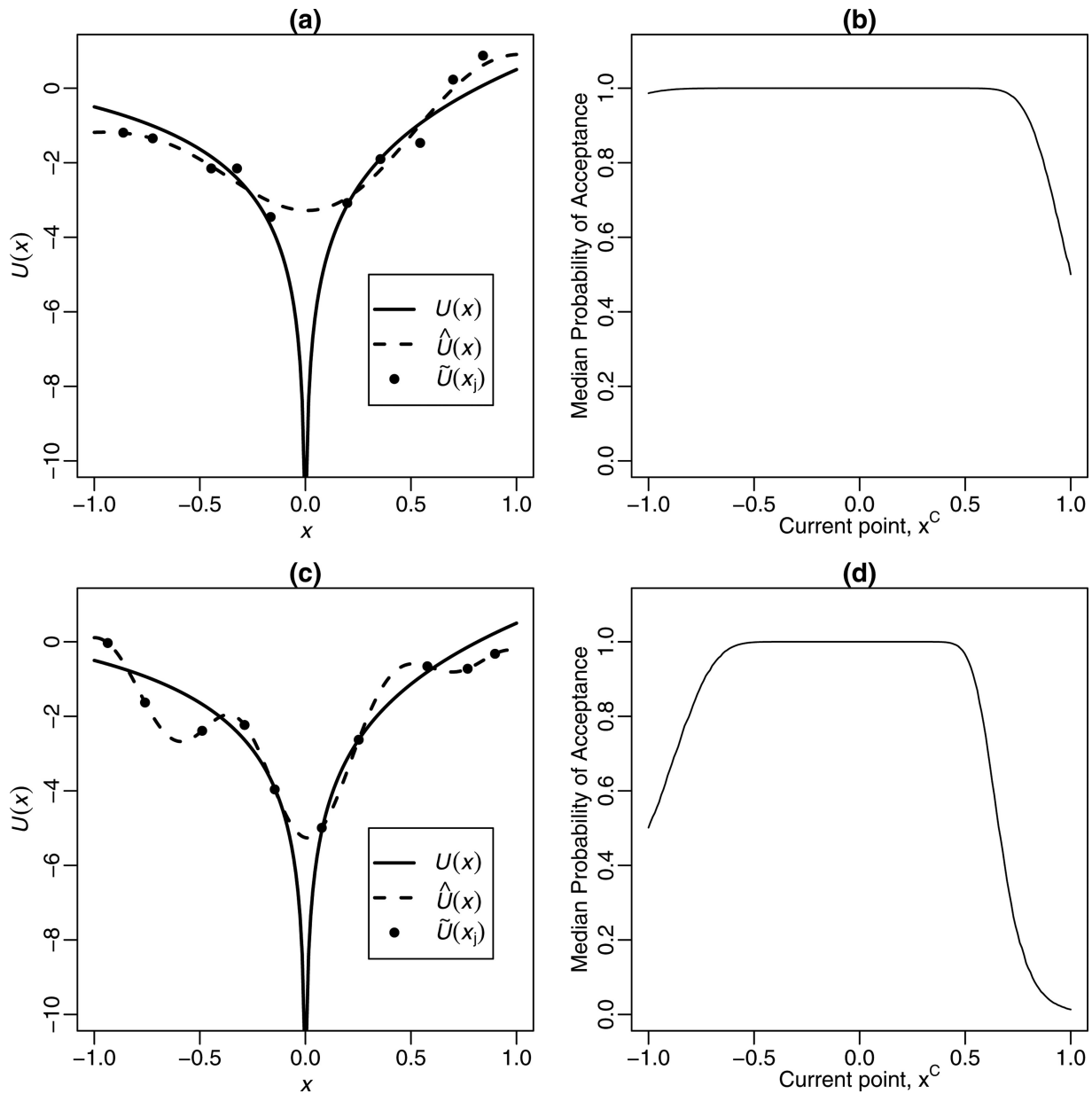
where  $\mathcal{I}(\beta; x)$  denotes the Fisher information. The expected utility is  $U(x) = 2 \log |x| + 0.5x$  and the optimal design is  $x^* = 1$ .

To simulate one iteration of Phase I of the ACE algorithm, we generate a coordinate-design  $\xi^1 = \{x^1, \dots, x^m\}$  as a Latin hypercube and, for each  $x^j$ , evaluate

$$\tilde{U}(x^j) = 2 \log |x^j| + \frac{x^j}{B} \sum_{i=1}^B \beta_i,$$

where  $\{\beta_i\}_{i=1}^B$  for  $B = 2$  is a sample from an  $N(0.5, 1)$  distribution. Figure 1(a) shows  $U(x)$  plotted against  $x$  with the points  $\{x^j, \tilde{U}(x^j)\}_{j=1}^m$  and the GP emulator  $\hat{U}(x)$  superimposed (Steps 2a, 2b, and 2c). Clearly  $\hat{U}(x)$  is maximized at  $x^\dagger = 1$ , and hence this candidate point should be compared to the current point  $x^C$  (Step 2d). Figure 1(b) shows the median posterior probability,  $p_I^\dagger$ , of accepting this candidate point against  $x^C$ , calculated from repeated calculation of (3) for multiple Monte Carlo samples. This probability is very close to one for nearly all values of  $x^C$  except for  $x^C \approx x^\dagger$ , where the probability reduces to  $1/2$ .

For a second coordinate-design,  $\xi^2$  (a different Latin hypercube), the results in Figure 1(c) and 1(d) are obtained. Here, the GP emulator could be viewed as inadequate with the estimate of  $\eta$  being too small, resulting in near interpolation of the  $\tilde{U}(x^j)$ . From Figure 1(c),  $\hat{U}(x)$  is maximized at  $x^\dagger = -1$  and hence this



**Figure 1.** Poisson example in Section 2.4. (a), (c) expected utility  $U(x)$  against  $x$ , with Monte Carlo evaluations  $\tilde{U}(x)$  at the coordinate-design points and GP emulator  $\hat{U}(x)$ ; (b), (d) median probability  $p_j^\dagger$  of accepting the candidate point against the current point,  $x^C$ . [Coordinate-designs are:  $\xi^1$  for (a), (b);  $\xi^2$  for (c), (d)].

becomes the candidate point. The median posterior acceptance probability, Figure 1(d), is now only close to one if  $\tilde{U}(x^C)$  is low, that is,  $|x^C| < 0.5$ . Crucially,  $x^\dagger$  will be rejected with high probability if  $x^C$  is close to the optimal design; at  $x^C = 1$ , the probability drops to zero.

### 3. Substantive Examples

The ACE algorithm is now used to find decision-theoretic Bayesian designs for three important cases: a compartmental model, (hierarchical) logistic regression, and dose-response under model-averaging. The designs are found for commonly used utility functions and, where possible, compared to existing results.

The supplementary material for this article provides a detailed vignette that demonstrates the use of the *acebayes*

package to find designs for the following examples, and R code to allow their straightforward reproduction.

#### 3.1 Utility Functions

In this section, we assess and compare designs found using variants on two utility functions, Shannon information gain (SIG) and negative squared error loss (NSEL). In practice, the form of the chosen utility function should be driven by the aims of the experiment and may often incorporate a cost function. We assume throughout that the model parameters can be expressed as  $\psi^T = (\theta^T, \gamma^T)$ , with  $\theta$  a  $p$ -vector of parameters of interest and  $\gamma$  a  $(P - p)$ -vector of nuisance parameters.

The SIG utility for  $\theta$  is given by

$$\begin{aligned} u^S(\theta, \gamma, \delta) &= \log \pi(\theta | \gamma, \delta) - \log \pi(\theta) \\ &= \log \pi(\gamma | \theta, \delta) - \log \pi(\gamma | \delta), \end{aligned} \quad (5)$$

where (5) follows from an application of Bayes' theorem and is often more useful for computation. A SIG-optimal design maximizes  $U^S(\delta) = E_{\psi, \mathbf{y}}[u^S(\boldsymbol{\theta}, \mathbf{y}, \delta)]$ . This is equivalent to maximizing the expected Kullback–Liebler distance between the marginal posterior and prior distributions of  $\boldsymbol{\theta}$ , and is also equivalent to minimizing the expected entropy of the posterior distribution for  $\boldsymbol{\theta}$ .

The NSEL utility for  $\boldsymbol{\theta}$  is given by

$$u^V(\boldsymbol{\theta}, \mathbf{y}, \delta) = - \sum_{w=1}^p [\theta_w - E(\theta_w | \mathbf{y}, \delta)]^2. \quad (6)$$

An NSEL-optimal design maximizes the expected utility  $U^V(\delta)$ , which is equivalent to minimizing the expectation of the trace of the posterior covariance matrix of  $\boldsymbol{\theta}$  with respect to the marginal distribution of  $\mathbf{y}$ .

### 3.1.1 Evaluating the Expected Utility via Numerical Approximation

For many statistical models, including most nonlinear models, evaluation of  $u^S(\boldsymbol{\theta}, \mathbf{y}, \delta)$  and  $u^V(\boldsymbol{\theta}, \mathbf{y}, \delta)$  requires numerical approximation. For given values of  $\mathbf{y}$  and  $\boldsymbol{\theta}$ , the components of (5) can be approximated as

$$\begin{aligned} \tilde{\pi}(\mathbf{y} | \boldsymbol{\theta}, \delta) &= \frac{1}{\tilde{B}} \sum_{b=1}^{\tilde{B}} \pi(\mathbf{y} | \boldsymbol{\theta}, \tilde{\mathbf{y}}_b, \delta), \\ \tilde{\pi}(\mathbf{y} | \delta) &= \frac{1}{\tilde{B}} \sum_{b=1}^{\tilde{B}} \pi(\mathbf{y} | \tilde{\boldsymbol{\theta}}_b, \tilde{\mathbf{y}}_b, \delta), \end{aligned}$$

where  $\{\tilde{\boldsymbol{\theta}}_b, \tilde{\mathbf{y}}_b\}_{b=1}^{\tilde{B}}$  is a size  $\tilde{B}$  random sample from the prior distribution of  $\boldsymbol{\psi} = (\boldsymbol{\theta}, \boldsymbol{\gamma})$ . These quantities can be incorporated into a nested, or double-loop, Monte Carlo approximation of  $U^S(\delta)$ :

$$\tilde{U}^S(\delta) = \frac{1}{B} \sum_{l=1}^B [\log \tilde{\pi}(\mathbf{y}_l | \boldsymbol{\theta}_l, \delta) - \log \tilde{\pi}(\mathbf{y}_l | \delta)],$$

with  $\{\mathbf{y}_l, \boldsymbol{\theta}_l\}_{l=1}^B$  a sample from the joint distribution of the response and parameters. Intuitively, the “inner sample” of size  $\tilde{B}$  is used to approximate the two marginal likelihoods in (5), the first marginal to  $\boldsymbol{\gamma}$  and the second to both  $\boldsymbol{\gamma}$  and  $\boldsymbol{\theta}$ , and the “outer sample” of size  $B$  is then used to approximate the expected utility with respect to the joint distribution of  $\mathbf{y}$  and  $\boldsymbol{\theta}$ . This approximation is biased for  $U^S(\delta)$  due to the bias in  $\log \tilde{\pi}(\mathbf{y} | \boldsymbol{\theta}, \delta)$  and  $\log \tilde{\pi}(\mathbf{y} | \delta)$ . However, under regularity conditions satisfied by most models of practical importance (Severini 2000, pp. 80–81), this bias is of order  $\tilde{B}^{-1}$  (Ryan 2003) and hence asymptotically negligible.

For NSEL,  $E(\theta_w | \mathbf{y}, \delta)$  in (6) can be approximated via importance sampling:

$$\tilde{E}(\theta_w | \mathbf{y}, \delta) = \frac{\sum_{b=1}^{\tilde{B}} \tilde{\theta}_{lw} \pi(\mathbf{y} | \tilde{\boldsymbol{\theta}}_b, \tilde{\mathbf{y}}_b, \delta)}{\sum_{b=1}^{\tilde{B}} \pi(\mathbf{y} | \tilde{\boldsymbol{\theta}}_b, \tilde{\mathbf{y}}_b, \delta)},$$

where  $\{\tilde{\boldsymbol{\theta}}_b, \tilde{\mathbf{y}}_b\}_{b=1}^{\tilde{B}}$  is a random sample from the prior distribution of  $\boldsymbol{\psi}$ , and  $\tilde{\theta}_{bw}$  is the  $w$ th element of  $\tilde{\boldsymbol{\theta}}_b$ . Hence, the following nested Monte Carlo approximation of the expected utility is

obtained:

$$\tilde{U}^V(\delta) = - \frac{1}{B} \sum_{l=1}^B \sum_{w=1}^p [\theta_{lw} - \tilde{E}(\theta_w | \mathbf{y}_l, \delta)]^2,$$

where  $\theta_{lw}$  is the  $w$ th element of  $\boldsymbol{\theta}_l$ . Here, the inner sample is used to approximate the posterior expectation, and the outer sample used to approximate the expected utility. Importance sampling has commonly been used to estimate posterior quantities for Bayesian design (see Ryan et al. 2016 and references therein), although the approximation of the expected utility will again be biased due to bias in  $\tilde{E}(\theta_w | \mathbf{y}, \delta)^2$ .

In the examples, we set  $\tilde{B} = B = 1000$  for the evaluation of  $\tilde{U}(\delta_i | \delta_{(i)}^C)$  in Step 2b of the ACE algorithm (chosen from practical experience). For the comparisons in Steps 2d and 4e, we set  $B = \tilde{B} = 20,000$ .

### 3.1.2 Evaluating the Expected Utility via Normal Approximation

The following approximations to  $U^S(\delta)$  and  $U^V(\delta)$  are commonly used (Atkinson, Donev, and Tobias 2007, chaps. 10, 18), justified via a normal approximation to the posterior distribution of  $\boldsymbol{\psi}$ :

$$\begin{aligned} \phi^S(\delta) &= -E_{\boldsymbol{\psi}}(\log |\mathbf{A}^T \mathcal{I}(\boldsymbol{\psi}; \delta)^{-1} \mathbf{A}|) \\ &= - \int_{\boldsymbol{\psi}} \log |\mathbf{A}^T \mathcal{I}(\boldsymbol{\psi}; \delta)^{-1} \mathbf{A}| \pi(\boldsymbol{\psi}) d\boldsymbol{\psi}, \\ \phi^V(\delta) &= -E_{\boldsymbol{\psi}}[\text{tr}\{\mathbf{A}^T \mathcal{I}(\boldsymbol{\psi}; \delta)^{-1} \mathbf{A}\}] \\ &= - \int_{\boldsymbol{\psi}} \text{tr}\{\mathbf{A}^T \mathcal{I}(\boldsymbol{\psi}; \delta)^{-1} \mathbf{A}\} \pi(\boldsymbol{\psi}) d\boldsymbol{\psi}, \end{aligned}$$

with  $\mathcal{I}(\boldsymbol{\psi}; \delta)$  the Fisher information matrix for  $\boldsymbol{\psi}$ , or an approximation thereof, and  $\mathbf{A} = [\mathbf{I}_p, \mathbf{0}_{p \times (p-p)}]^T$  with  $\mathbf{I}_p$  the  $p \times p$  identity matrix and  $\mathbf{0}_{a \times b}$  the  $a \times b$  zero matrix. Designs that maximize  $\phi^S$  and  $\phi^V$  are sometimes referred to as pseudo-Bayesian  $D$ - and  $A$ -optimal designs, respectively. Note that these expressions also result from taking expectations of the utility functions

$$\begin{aligned} u^D(\boldsymbol{\psi}, \mathbf{y}, \delta) &= -\log |\mathbf{A}^T \mathcal{I}(\boldsymbol{\psi}; \delta)^{-1} \mathbf{A}|, \\ u^A(\boldsymbol{\psi}, \mathbf{y}, \delta) &= -\text{tr}\{\mathbf{A}^T \mathcal{I}(\boldsymbol{\psi}; \delta)^{-1} \mathbf{A}\}, \end{aligned}$$

which do not depend on  $\mathbf{y}$ . Unbiased Monte Carlo approximations to  $\phi^S(\delta)$  and  $\phi^V(\delta)$  can be obtained via sampling from the prior distribution for  $\boldsymbol{\psi}$ :

$$\begin{aligned} \hat{\phi}^S(\delta) &= -\frac{1}{B} \sum_{l=1}^B \log |\mathbf{A}^T \mathcal{I}(\boldsymbol{\psi}_l; \delta)^{-1} \mathbf{A}|, \\ \hat{\phi}^V(\delta) &= -\frac{1}{B} \sum_{l=1}^B \text{tr}\{\mathbf{A}^T \mathcal{I}(\boldsymbol{\psi}_l; \delta)^{-1} \mathbf{A}\}. \end{aligned}$$

For comparison of designs, the  $D$ -efficiency of design  $\delta_1$  relative to design  $\delta_2$  is defined as

$$\text{Eff}_D(\delta_1, \delta_2) = 100 \times \exp \left\{ [\hat{\phi}^S(\delta_1) - \hat{\phi}^S(\delta_2)] / p \right\}. \quad (7)$$

### 3.2 Compartmental Model

Compartmental models are applied in pharmacokinetics to study how materials flow through an organism, and have been used extensively to demonstrate optimal design methodology (Atkinson et al. 1993; Gotwalt, Jones, and Steinberg 2009). The archetypal design problem is to choose  $n$  sampling times  $\delta = (t_1, \dots, t_n)^T$ , in hours, at which to measure the concentration in a subject of a previously administered drug. Here, concentration is modeled as  $y_i \sim N(a(\theta)\mu(\theta; t_i), \sigma^2 b(\theta; t_i))$ , where  $\theta = (\theta_1, \theta_2, \theta_3)^T$  are the parameters of interest,  $\sigma^2 > 0$  is a nuisance parameter,  $a(\cdot)$  and  $b(\cdot; \cdot)$  are application-dependent functions, and  $\mu(\theta; t_i) = \exp(-\theta_1 t_i) - \exp(-\theta_2 t_i)$ .

For this problem, Ryan et al. (2014) assumed that

$$a(\theta) = \frac{400\theta_2}{\theta_3(\theta_2 - \theta_1)}, \quad b(\theta; t_i) = \left(1 + \frac{a(\theta)^2 \mu(\theta; t_i)^2}{10}\right),$$

$$\sigma^2 = 0.1,$$

and found designs using the SIG utility function. Independent log-normal prior distributions were assumed for the elements of  $\theta$  with, on the log scale, each having common variance 0.05 and expectations  $\log(0.1)$ ,  $\log(1)$ , and  $\log(20)$  for  $\theta_1$ ,  $\theta_2$ , and  $\theta_3$ , respectively. These authors also incorporated the constraint  $\max_{s,t=1,\dots,n} |t_s - t_t| \geq 0.25$ , that is, that sampling times must be at least 15 min apart. It is straightforward to incorporate this constraint into design search using the ACE algorithm. In Step 2d,  $\hat{U}(\delta_i | \delta_{(i)}^c)$  is maximized over a set  $\mathcal{D}_i$  that satisfies the constraint. Phase II of the ACE algorithm is then omitted as replicated sampling times are not permitted.

Ryan et al. (2014) restricted their search for an SIG-optimal design to the class of designs defined via a dimension reduction scheme (DRS) that set the  $n$  sampling times to scaled percentiles of a Beta( $\alpha_1, \alpha_2$ ) distribution. Hence, the design problem was reduced to selecting two parameters,  $\alpha_1$  and  $\alpha_2$ . The Müller (1999) simulation algorithm was used to sample from an artificial posterior distribution for  $\alpha_1, \alpha_2$ , with unnormalized density equal to the integrand in (1). The chosen design was then the scaled quantiles from the Beta distribution obtained from using the posterior modal values of  $\alpha_1$  and  $\alpha_2$ .

We compare this design with three designs found from ACE: (i) an SIG-optimal design; (ii) a pseudo-Bayesian  $D$ -optimal design; and (iii) an optimal choice of  $\alpha_1, \alpha_2$  for the Beta DRS. For this latter design, the sampling times are given by  $t_j = 24 \times Q(\frac{j}{n+1}; \alpha_1, \alpha_2)$ , with  $Q(r; \alpha_1, \alpha_2)$  the  $r$ th quantile of the Beta( $\alpha_1, \alpha_2$ ) distribution. In Step 2d of the ACE algorithm, the sets  $\mathcal{D}_1$  and  $\mathcal{D}_2$  are given by

$$\mathcal{D}_1 = \left\{ x \in \mathbb{R}^+ : \min_{j=1,\dots,n-1} \left| Q\left(\frac{j}{n+1}, x, \alpha_2\right) - Q\left(\frac{j+1}{n+1}, x, \alpha_2\right) \right| > \frac{0.25}{24} \right\},$$

$$\mathcal{D}_2 = \left\{ x \in \mathbb{R}^+ : \min_{j=1,\dots,n-1} \left| Q\left(\frac{j}{n+1}, \alpha_1, x\right) - Q\left(\frac{j+1}{n+1}, \alpha_1, x\right) \right| > \frac{0.25}{24} \right\}.$$

For the SIG- and  $D$ -optimal designs, Figure 2(a) and 2(b) shows trace plots of the approximate expected utility at each iteration of the algorithm. Approximate convergence is demonstrated, for both utility functions, through each run of the algorithm resulting in a similar value of  $\tilde{U}(\delta)$  after a relatively small number of iterations. Convergence is, however, achieved more quickly for pseudo-Bayesian  $D$ -optimality, which does not require approximation of posterior quantities. This criterion also displays greater consistency in the final approximated expected utility between runs of the algorithm.

The sampling times for the four designs, shown in Figure 2(c), indicate that the designs using dimension-reduction do not display the clustering of points evident in the SIG and pseudo-Bayesian  $D$ -optimal designs. The boxplots in Figure 2(d), from 20 evaluations of  $\tilde{U}^S(\delta^*)$  ( $B = 20,000$ ) for each design, confirm that larger approximate expected utilities are obtained, up to a 5% improvement, when DRS is not used. Here, the pseudo-Bayesian  $D$ -optimal design provides a good approximation to the SIG-optimal design.

The DRS design found from ACE outperforms the Ryan et al. (2014) design. To explore this result further, the expected utility surface was investigated as a function of  $\alpha_1$  and  $\alpha_2$  by sampling 40,000  $(\alpha_1, \alpha_2)$  pairs from  $[0, 5]^2$  and evaluating  $\tilde{U}^S(\delta)$  for each pair. The resulting expected utility surface is shown in Figure 2(e), where  $\tilde{U}^S(\delta) = 0$  for parameter pairs that do not satisfy the 15 min constraint. Both methods identify the relatively small region of high expected utility; the sampling-based algorithm (Müller 1999; Ryan et al. 2014) fails to identify the optimum point within this region.

### 3.3 Logistic Regression in Four Factors

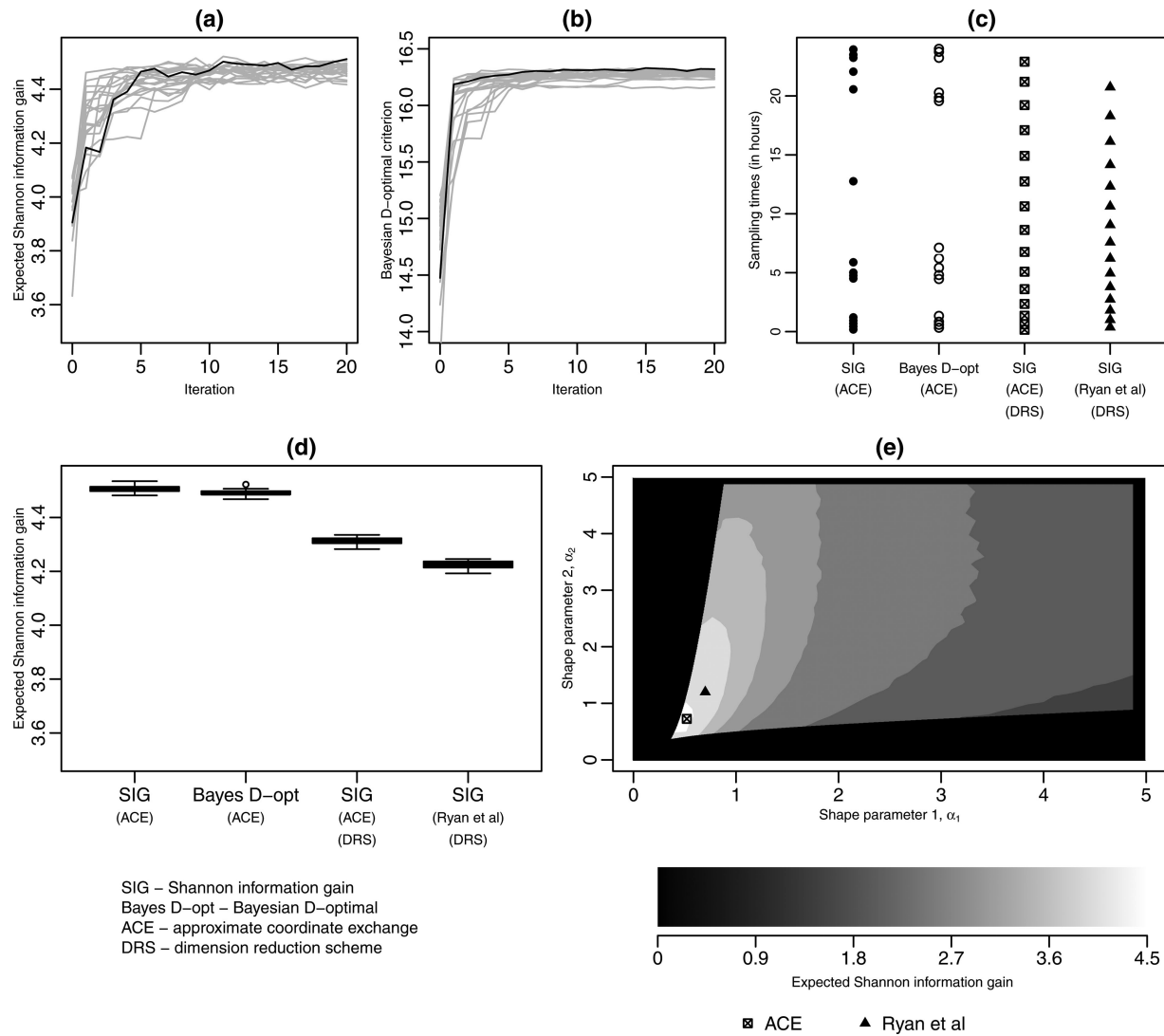
Fully-Bayesian design for multi-variable logistic regression has not appeared in the literature, although Hamada et al. (2001) found an SIG-optimal design for a single-variable model and Woods et al. (2006) were the first to find multi-variable pseudo-Bayesian  $D$ -optimal designs. Here, we find designs for a first-order logistic regression model in four variables where the response is measured for  $G$  groups of  $n_g$  runs, that is,  $n = Gn_g$ . Let  $y_{st} \sim \text{Bernoulli}(\rho_{st})$  be the  $t$ th response from the  $s$ th group ( $s = 1, \dots, G$ ;  $t = 1, \dots, n_g$ ), with

$$\log\left(\frac{\rho_{st}}{1 - \rho_{st}}\right) = \beta_0 + \omega_{s0} + (\beta_1 + \omega_{s1})x_{1st} + (\beta_2 + \omega_{s2})x_{2st} \\ + (\beta_3 + \omega_{s3})x_{3st} + (\beta_4 + \omega_{s4})x_{4st} \\ = \mathbf{x}_{st}^T (\boldsymbol{\beta} + \boldsymbol{\omega}_s),$$

where  $\boldsymbol{\beta} \in \mathbb{R}^5$  are the parameters of interest, and  $\boldsymbol{\omega}_s \in \mathbb{R}^5$  ( $s = 1, \dots, G$ ) are the group-specific nuisance parameters (or “random effects”). Let  $\mathbf{X} = (\mathbf{X}_1^T \dots \mathbf{X}_G^T)^T$  be the  $n \times 5$  model matrix where  $\mathbf{X}_s$  is the  $n_g \times 5$  matrix with  $t$ th row given by  $\mathbf{x}_{st}^T$ . The design matrix  $\mathbf{D}$  is formed as the last four columns of  $\mathbf{X}$ ,  $\boldsymbol{\delta} = \text{vec}(\mathbf{D})$  has length  $q = 4n$ , and  $\mathcal{D}_i = [-1, 1]$  for  $i = 1, \dots, q$ .

The following independent prior distributions for each element of  $\boldsymbol{\beta}$  are assumed:





**Figure 2.** (a), (b) Trace plots of  $\tilde{U}(\delta^C)$  for each iteration of the ACE algorithm for SIG and pseudo-Bayesian  $D$ -optimality utilities, respectively; in each plot, the black line shows the trace of the expected utility for the best design; (c) designs found from the ACE algorithm: unrestricted SIG-optimal, pseudo-Bayesian  $D$ -optimal, Beta DRS SIG-optimal, together with the Ryan et al. (2014) Beta DRS SIG-optimal designs; (d) boxplots for 20 evaluations of  $\tilde{U}^S(\delta^*)$  for designs from these four methodologies; (e) approximate expected utility surface for SIG as a function of the Beta DRS parameters; parameter values corresponding to the Ryan et al. (2014) and the ACE DRS designs are marked.

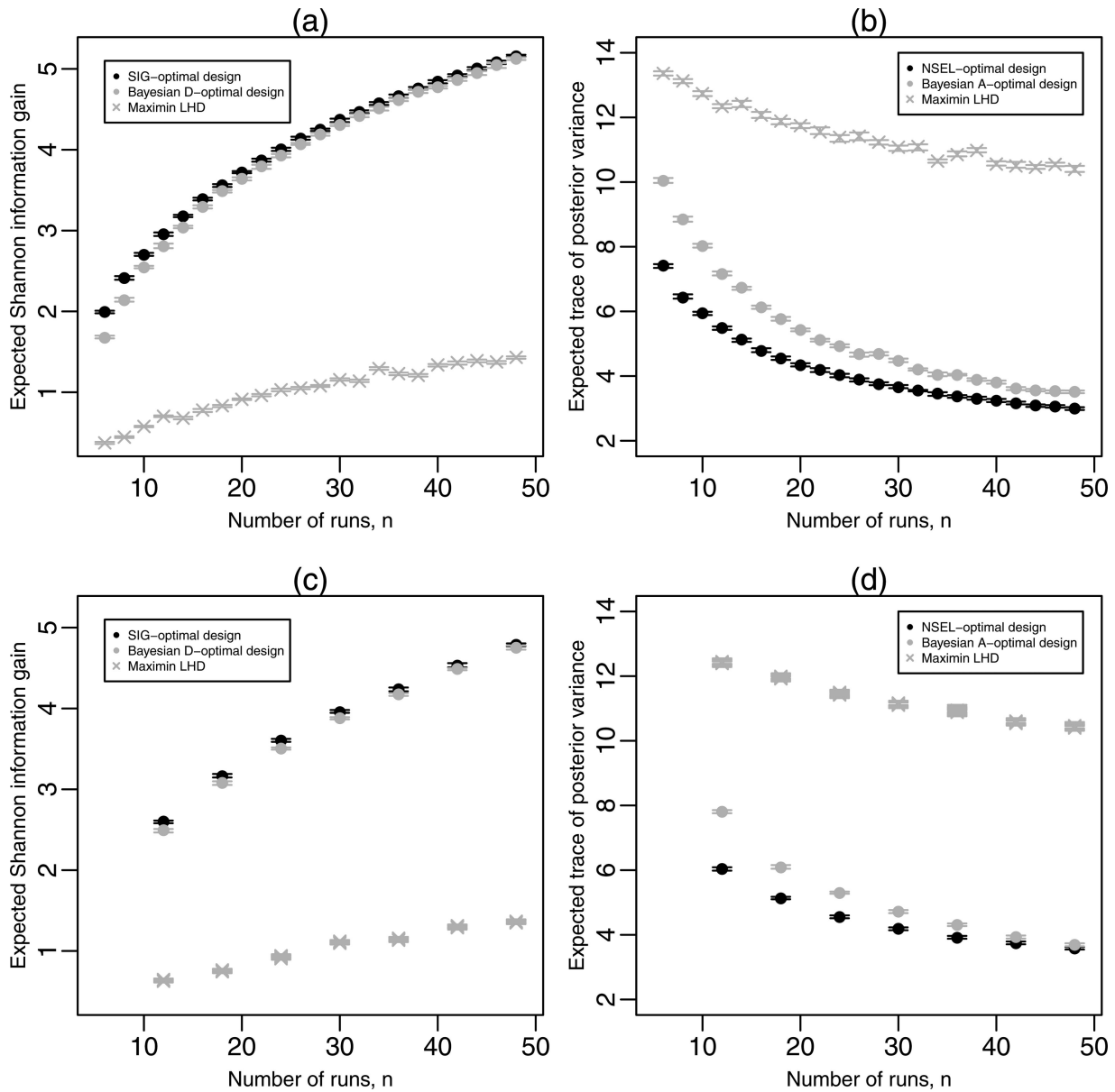
$$\begin{aligned} \beta_0 &\sim U[-3, 3], & \beta_1 &\sim U[4, 10], & \beta_2 &\sim U[5, 11], \\ \beta_3 &\sim U[-6, 0], & \beta_4 &\sim U[-2.5, 3.5]. \end{aligned} \quad (8)$$

We find designs for two different prior distributions for each  $\omega_s$  ( $s = 1, \dots, G$ ): (i) a prior point mass at  $\omega_s = \mathbf{0}$  for all  $s$ , resulting in standard logistic regression with homogenous groups; (ii) a hierarchical prior distribution in which the elements of  $\omega_s$  are independent and identically distributed as  $\omega_{sr} \sim U[-\lambda_r, \lambda_r]$ , for  $r = 0, \dots, 4$ , with  $\lambda_r > 0$  unknown and having triangular prior density  $\pi(\lambda_r) = \frac{2(L_r - \lambda_r)}{L_r^2}$  with  $(L_0, \dots, L_4) = (3, 3, 3, 1, 1)$ .

### 3.3.1 Logistic Regression With Homogenous Groups

We use ACE to find designs that maximize the SIG and NSEL expected utilities for homogenous logistic regression with  $\omega_s = 0$  and  $n = 6, \dots, 48$ . For comparison, we also find pseudo-Bayesian  $D$ - and  $A$ -optimal designs. We also compare to maximin Latin hypercube (LH) designs (Morris and Mitchell 1995). For this example, the starting designs for the algorithm were

a locally  $D$ -optimal design (for SIG and Bayesian  $D$ ) and a locally  $A$ -optimal design (for NSEL and Bayesian  $A$ ), found from ACE via maximization of  $\psi^S(\delta)$  or  $\psi^V(\delta)$ , respectively, using a point prior distribution for each parameter with support at the mean of each prior distribution in (8). Figure 3 presents results (minimum, mean, maximum) of 20 evaluations of (a)  $\tilde{U}^S(\delta)$  for the SIG-optimal, Bayesian  $D$ -optimal, and maximin LH designs, and (b)  $-\tilde{U}^V(\delta)$  for the NSEL-optimal, Bayesian  $A$ -optimal, and maximin LH designs, using  $B = 20,000$  Monte Carlo samples. For small  $n$ , on average there are substantial differences in expected utility between the fully Bayesian and pseudo-Bayesian designs, with the SIG-optimal design having expected Shannon information gain up to 20% larger than the Bayesian  $D$ -optimal design and the NSEL-optimal design having expected trace of the posterior covariance matrix up to 27% smaller than the Bayesian  $A$ -optimal design. For both SIG and NSEL, as  $n$  increases, the difference in expected utility between these designs and the pseudo-Bayesian designs decreases. For



**Figure 3.** Results from 20 evaluations of (a)  $\bar{U}^S(\delta)$  for SIG-optimal, pseudo-Bayesian  $D$ -optimal, and maximin Latin hypercube designs, and (b)  $-\bar{U}^V(\delta)$  for NSEL-optimal, pseudo-Bayesian  $A$ -optimal, and maximin Latin hypercube designs, for homogenous logistic regression; (c) and (d) show the same evaluations for hierarchical logistic regression. For the latter two plots, for each value of  $n$ , 20 different random assignments are made of the points of the Latin hypercube design to the  $G$  groups, and each resulting design is evaluated 20 times. For each design, the central plotting symbol denotes the mean expected Shannon information gain or expected average posterior variance, with the two horizontal lines denoting the minimum and maximum of these quantities.

SIG, these findings agree with asymptotic results on the convergence, under certain regularity conditions, of the posterior distribution to a normal distribution (see, e.g., Gelman et al. 2014, pp. 585–588). The maximin LH designs, which are model-free space-filling designs, perform poorly under both SIG and NSEL utilities and are not competitive with the model-based designs.

As there are no comparable results on fully-Bayesian design for multi-variable logistic regression in the literature, we compare the pseudo-Bayesian  $D$ -optimal designs for  $n = 16$  and  $n = 48$  found from ACE with designs from the approach of Gotwalt, Jones, and Steinberg (2009). We independently implemented the methodology of these authors to obtain designs for  $n = 16$  and  $n = 48$ ; we also compare to the  $n = 16$  run design published by Gotwalt, Jones, and Steinberg (2009). For each of these three designs, we calculated the average of  $D$ -efficiency (7) over 20 Monte Carlo approximations (each with

$B = 20,000$ ) relative to the appropriately sized design from ACE. The published 16-run design has average efficiency of 82%; the designs from our implementation perform similarly to the ACE designs, with average efficiencies of 99.9% and 101.3% for  $n = 16$  and  $n = 48$ , respectively.

### 3.3.2 Hierarchical Logistic Regression

For hierarchical logistic regression, we again find SIG-optimal and NSEL-optimal designs, along with pseudo-Bayesian  $D$ - and  $A$ -optimal designs using an approximation to the Fisher information (Pawitan 2001, p. 467). We set  $n_g = 6$  and  $G = 2, \dots, 8$ , leading to  $n = 12, 14, \dots, 48$ . To reduce the computational burden,  $B = 1000$  was used in Step 4e to find SIG-optimal designs. Previous research has found pseudo-Bayesian  $D$ -optimal designs for smaller numbers of variables and group sizes (Waite and Woods 2015).

Figure 3(c) and 3(d) shows results from 20 evaluations of  $\tilde{U}^S(\delta)$  and  $-\tilde{U}^V(\delta)$  for the SIG-optimal and pseudo-Bayesian D-optimal designs, and the NSEL-optimal and pseudo-Bayesian A-optimal designs, respectively. Again, the performances of maximin LH designs are included for reference (see figure caption for details). A comparison with Figure 3(a) and 3(b), respectively, shows lower expected gains in Shannon information and higher expected posterior variance for the hierarchical logistic regression model due to additional uncertainty introduced by the group-specific parameters. As with designs for homogenous logistic regression, the difference in expected utility between the pseudo-Bayesian designs and the fully-Bayesian designs decreases as  $n$  increases, and the LH designs perform poorly.

### 3.4 Binomial Regression Under Model Uncertainty

Uncertainty over the choice of statistical model  $\pi(\mathbf{y}, \boldsymbol{\psi} | \delta)$  is common in practice, and has been addressed in pseudo-Bayesian design for generalized linear models by Woods et al. (2006). To demonstrate Bayesian optimal design under model uncertainty, we find follow-up designs for the beetle mortality study of Bliss (1935), a common example used to illustrate binomial regression. In the original dataset, 481 beetles were each administered one of eight different doses (in mg/L) of carbon disulfate. We broadly follow the case study analysis of O'Hagan and Forster (2004, pp. 423–433), who reproduced the data, and assume interest lies in providing a model-averaged posterior distribution of the lethal dose 50 (LD50), the dose required to achieve 50% mortality.

We assume that the binary indicator of death for each beetle is an independent Bernoulli random variable. The number,  $y_k$ , of deaths from dose  $x_k$  is modeled as  $y_k \sim \text{Binomial}(n_k, \rho_k)$ , where  $\rho_k$  is the probability of death for the  $k$ th dose, which was administered to  $n_k$  beetles,  $\sum_{k=1}^n n_k = 481$ . We denote the link function by  $g(\rho_k) = \eta_k$ , with  $\eta_k$  the linear predictor and consider six models formed by the Cartesian product of three link functions and two linear predictors: the logit,  $g(\rho_k) = \log\{\rho_k/(1 - \rho_k)\}$ , the c-log-log,  $g(\rho_k) = \log\{-\log(1 - \rho_k)\}$ , and the probit,  $g(\rho_k) = \Phi^{-1}\{\rho_k\}$ , with  $\Phi\{\cdot\}$  the standard normal distribution function; and first-order ( $\eta_i = \beta_0 + \beta_1 x_i$ ) and second-order ( $\eta_i = \beta_0 + \beta_1 x_i + \beta_2 x_i^2$ ) linear predictors.

Let  $u \in \mathcal{U} = \{1, \dots, 6\}$  denote the model indicator (see Table 1) and let  $\boldsymbol{\beta}_u$  denote the vector of regression parameters under model  $u$ . LD50 is then given by

$$\text{LD}(\boldsymbol{\beta}_u)$$

**Table 1.** Approximate posterior model probabilities,  $\pi(u | \mathbf{y})$ , for the beetle mortality data.

$u$	Link function	Linear predictor	$\pi(u   \mathbf{y})$
1	Logit	1st order	0.0216
2	Logit	2nd order	0.0686
3	C-log-log	1st order	0.7580
4	C-log-log	2nd order	0.0612
5	Probit	1st order	0.0304
6	Probit	2nd order	0.0602

$$= \begin{cases} \frac{w - \beta_0}{\beta_1} & \text{for } u = 1, 3, 5 \\ & \text{(first-order linear predictor)} \\ \frac{-\beta_1 + \sqrt{\beta_1^2 - 4\beta_2(\beta_0 - w)}}{2\beta_2} & \text{otherwise,} \end{cases}$$

where  $w = \log\{-\log(0.5)\}$  for the c-log-log link function, and 0 otherwise. We use unit information prior distributions (Ntzoufras, Dellaportas, and Forster 2003) for  $\boldsymbol{\beta}_u | u$  under each model and set  $\pi(u) = 1/6$  for  $u = 1, \dots, 6$ . The posterior model probabilities for each model are approximated using importance sampling to evaluate the marginal likelihood of each model, and given in Table 1. Samples from the posterior distribution of the model parameters are generated for each of the six models using the Metropolis–Hastings algorithm, and then weighted by  $\pi(u | \mathbf{y})$  to produce a sample from the joint posterior distribution  $\boldsymbol{\beta}_u, u | \mathbf{y}$  of regression parameters and model indicator. A sample from the model-averaged posterior distribution of LD50 can be obtained by evaluating  $\text{LD}(\boldsymbol{\beta}_u)$  for each sampled parameter vector.

We consider the design of a follow-up experiment using a further  $n_0$  (potentially new) doses. Each dose is to be administered to  $n_{0k_0}$  beetles ( $k_0 = 1, \dots, n_0$ ) and, in each group, the number,  $y_{0k_0}$ , of beetles that die is recorded. Let  $\mathbf{y}_0$  be the  $n_0 \times 1$  vector of the numbers of beetles that die in the follow-up experiment. We assume that  $n_{0k_0}$  is unknown and adopt a  $\text{Poisson}(\lambda)$  prior distribution. Hence  $y_{0k_0} \sim \text{Poisson}(\lambda \rho_{k_0})$ . We choose  $\lambda = 60$ , consistent with the values of  $n_k$  in the original dataset, and find designs for  $n_0 = 1, \dots, 10$  to estimate the value of LD50 under the NSEL utility function by maximizing

$$U^V(\delta) = - \sum_{u=1}^6 \pi(u | \mathbf{y}) \int_{\mathcal{Y}} \int_{\mathcal{B}_u} [\text{LD}(\boldsymbol{\beta}_u) - \text{E}(\text{LD}(\boldsymbol{\beta}_u) | \mathbf{y}_0, \mathbf{y}, \delta)]^2 \times \pi(\boldsymbol{\beta}_u, \mathbf{y}_0 | u, \mathbf{y}) d\boldsymbol{\beta}_u d\mathbf{y}_0,$$

where design  $\delta$  is the  $n_0 \times 1$  vector of doses and  $\mathcal{B}_u$  is the parameter space for model  $u$ . For the purposes of design and modeling, we assume that  $\delta_i \in \mathcal{D}_i = [-1, 1]$  for all  $i = 1, \dots, n_0$  but transform the doses to the original scale [1.6907, 1.8839] for the presentation of results.

We can approximate  $U^V(\delta)$  by

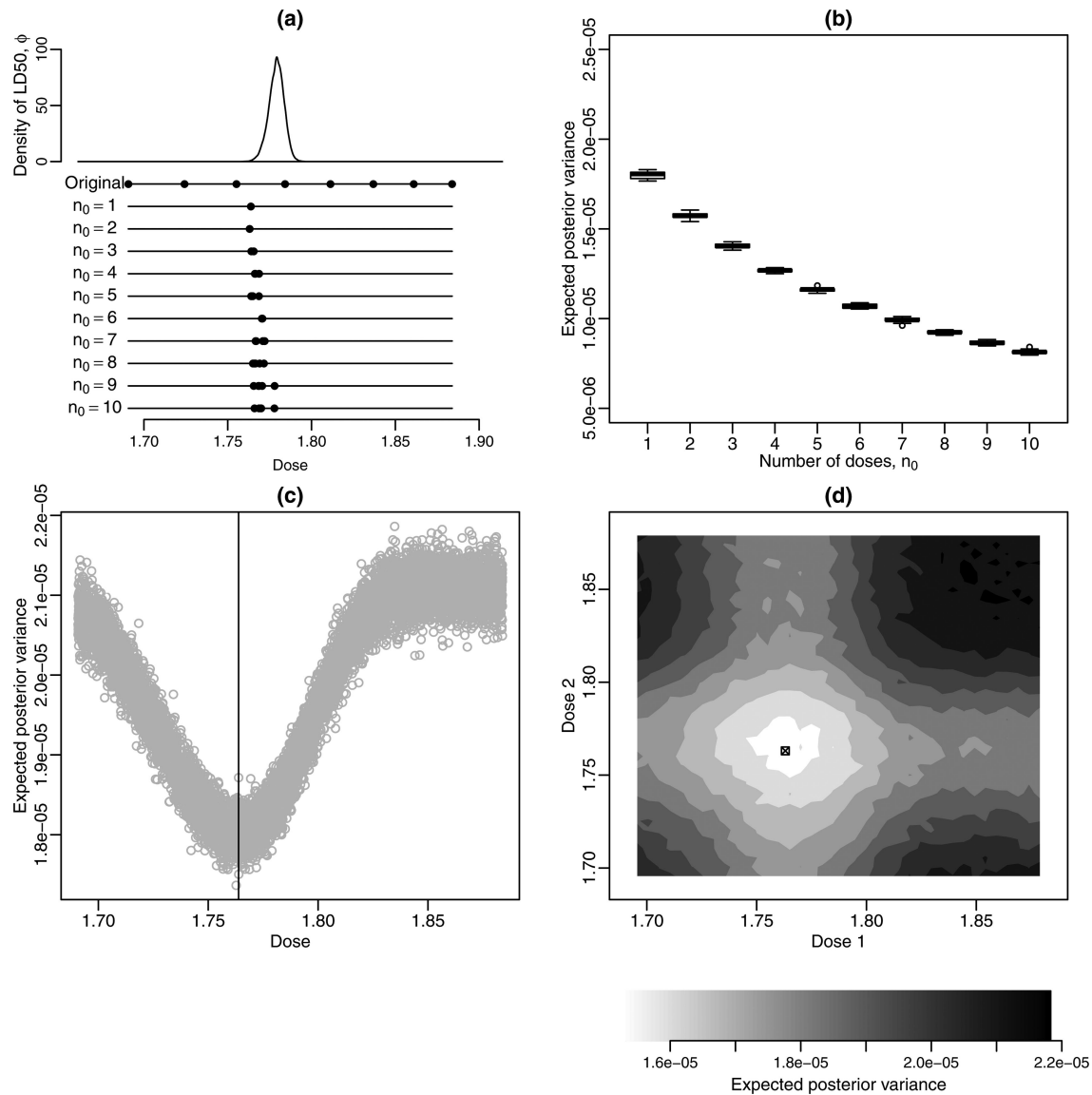
$$\tilde{U}^V(\delta) = -\frac{1}{B} \sum_{l=1}^B [\text{LD}(\boldsymbol{\beta}_{ul}) - \hat{\text{E}}(\text{LD}(\boldsymbol{\beta}_u) | \mathbf{y}_{0l}, \mathbf{y}, \delta)]^2,$$

where  $\{\boldsymbol{\beta}_{ul}, u_l, \mathbf{y}_{0l}\}_{l=1}^B$  is a random sample from the joint distribution with density  $\pi(\boldsymbol{\beta}_u, u, \mathbf{y}_0 | \mathbf{y})$ , and

$$\hat{\text{E}}(\text{LD}(\boldsymbol{\beta}_m) | \mathbf{y}_0, \mathbf{y}, \delta) = \frac{\sum_{b=1}^{\tilde{B}} \text{LD}(\tilde{\boldsymbol{\beta}}_{ub}) \pi(\mathbf{y}_0 | \tilde{\boldsymbol{\beta}}_{ub}, \tilde{m}_b)}{\sum_{b=1}^{\tilde{B}} \pi(\mathbf{y}_0 | \tilde{\boldsymbol{\beta}}_{ub}, \tilde{m}_b)},$$

where  $\{\tilde{\boldsymbol{\beta}}_{ub}, \tilde{m}_b\}_{b=1}^{\tilde{B}}$  is a random sample generated from the joint distribution with density  $\pi(\boldsymbol{\beta}_u, u | \mathbf{y})$ .

Figure 4 summarizes the results from the ACE algorithm. The doses in the NSEL-optimal design lie in the lower tail of the (original) posterior distribution of LD50 for all values of  $n_0$ , see Figure 4(a). For  $n_0 > 1$ , the doses are concentrated near



**Figure 4.** (a) Posterior density for LD50, the original experimental doses and optimal doses (in mg/L) for each value of  $n_0$ ; (b) boxplots of 20 evaluations of  $-\tilde{U}^V(\delta^*)$  for each  $n_0$  for the NSEL-optimal designs; (c) negative approximate expected utility  $-\tilde{U}^V(\delta)$  against dose for  $n_0 = 1$ ; the vertical line indicates  $\delta^*$ . (d) negative approximate expected utility  $-\tilde{U}^V(\delta)$  against dose for  $n_0 = 2$ ;  $\boxtimes$  indicates  $\delta^*$ .

a single point (1.77), for example, four replicate points occur for  $n_0 = 10$ . The approximate expected posterior variance of LD50,  $-\tilde{U}^V(\delta)$ , rapidly decreases as  $n_0$  is initially increased from 1, see Figure 4(b); the rate of decrease slows as  $n_0$  becomes larger.

To further investigate the selected designs, the expected utility surface and the performance of the ACE algorithm, we randomly generated 10,000 designs for  $n_0 = 1$  and  $n_0 = 2$  uniformly from  $\mathcal{D}_1$  and  $\mathcal{D}_1^2$ , respectively. For each design, we evaluate  $-\tilde{U}^V(\delta)$  and plot against dose; see Figure 4(c) for  $n_0 = 1$  and Figure 4(d) for  $n_0 = 2$ . The NSEL design identified by ACE is marked in each plot and, for both values of  $n_0$ , the minimum negative expected utility is achieved. The variance of the original model-averaged posterior distribution for LD50 is  $2.10 \times 10^{-5}$ . Hence for both  $n_0 = 1$  and  $n_0 = 2$ , it is clear that choosing a design composed of only very high or low doses would have resulted in a negligible expected reduction in variance.

## 4. Discussion and Future Work

The ACE methodology proposed in this article provides a step-change in the nature and complexity of statistical models and experiments for which Bayesian designs can be obtained. It may be used to find decision-theoretic designs whenever it is possible to simulate values from the prior distribution of the model parameters and responses from the statistical model. The combination of emulating an approximation to the expected utility and the coordinate exchange algorithm has allowed much larger problems to be tackled than was previously possible, both greater numbers of runs and more controllable variables. The algorithm also matches or exceeds the performance of existing approaches for smaller problems, and offers a clear advantage for design selection over the application of a dimension reduction scheme. The new designs made possible by this methodology also allow previously impossible benchmarking of designs from asymptotic approximations.



As presented, ACE can be applied to numerous important practical problems using the available R package. We have applied, or are in the process of applying, ACE to problems from chemical development and biological science. There are also a variety of extensions that could be made to ACE to increase its computational efficiency and applicability. We now highlight a few of these areas.

In ongoing work, we are extending and applying the methodology to find designs for statistical models where the likelihood function is only available numerically as the output from an expensive computer code (see also Huan and Marzouk 2013). Such models include those described by the solution to a system of nonlinear differential equations, which are increasingly studied in the field of uncertainty quantification (e.g., Chkrebti et al. 2015).

Convergence of the algorithm may be improved through a reparameterization of the design to remove dependencies between coordinates (e.g., Fletcher 1987, p. 19) that can be evident in efficient designs for some models. Such dependencies could be identified through pilot runs of the algorithm or by studying properties of pseudo-Bayesian designs. Additionally, the computational burden of the algorithm could be further reduced by employing alternative approaches to perform each one-dimensional optimization step in the algorithm. For example, a sequential strategy could use an expected improvement criterion modified for stochastic responses (e.g., Pichney et al. 2013).

Alternative strategies could also be adopted for the approximation of the expected utility. Zero-variance Monte Carlo (Ripley 1987, pp. 129–132; Mira, Solgi, and Imparato 2013) could be used to reduce the variance of the Monte Carlo estimator through the introduction of negative correlations via antithetic variables. Combining deterministic approximations, such as expectation propagation, with Monte Carlo methods would remove the need for nested simulation and may work well for nonlinear regression models with normal prior distributions.

## Supplementary Materials

The supplementary material includes the designs discussed in the article, and documentation and code to reproduce all the examples. The R package *acebayes* that implements the ACE algorithm is available on CRAN (<https://cran.r-project.org/package=acebayes>).

## Acknowledgments

This work was supported by the U.K. Engineering and Physical Sciences Research Council through Fellowship EP/J018317/1 for D.C. Woods. The authors thank the editor, an associate editor, and two reviewers for comments that improved the article, and the participants at the “Bayesian Optimal Design of Experiments” workshop (Brisbane, Australia, December 2015; <http://www.bode2015.wordpress.com>) for useful discussions on extensions and future work. The authors acknowledge the use of the IRIDIS High Performance Computing Facility, and associated support services at the University of Southampton, in the completion of this work.

## References

Amzal, B., Bois, F. Y., Parent, E., and Robert, C. (2006), “Bayesian Optimal Design via Interacting Particle Systems,” *Journal of the American Statistical Association*, 101, 773–785. [459]

- Atkinson, A., Donev, A., and Tobias, R. (2007), *Optimum Experimental Design, with SAS*, Oxford: Oxford University Press. [459,463]
- Atkinson, A. C., Chaloner, K., Herzberg, A. M., and Juritz, J. (1993), “Experimental Designs for Properties of a Compartmental Model,” *Biometrics*, 49, 325–337. [464]
- Bastos, L. S., and O’Hagan, A. (2009), “Diagnostics for Gaussian Process Emulators,” *Technometrics*, 51, 425–438. [461]
- Bliss, C. I. (1935), “The Calculation of the Dosage-Mortality Curve,” *Annals of Applied Biology*, 22, 134–167. [467]
- Chaloner, K., and Larntz, K. (1989), “Optimal Bayesian Design Applied to Logistic Regression Experiments,” *Journal of Statistical Planning and Inference*, 21, 191–208. [459]
- Chaloner, K., and Verdinelli, I. (1995), “Bayesian Experimental Design: A Review,” *Statistical Science*, 10, 273–304. [458]
- Chkrebti, O. A., Campbell, D. A., Calderhead, B., and Girolami, M. (2015), “Bayesian Solution Uncertainty Quantification for Differential Equations” (with discussion), *Bayesian Analysis*, 11, 1239–1299. [469]
- Dean, A., Morris, M., Stufken, J., and Bingham, D. (eds.) (2015), *Handbook of Design and Analysis of Experiments*, Boca Raton, FL: CRC Press. [458]
- Fletcher, R. (1987), *Practical Methods of Optimization* (2nd ed.), Chichester: Wiley. [469]
- Gelman, A., Carlin, J., Stern, H., Dunson, D., Vehtari, A., and Rubin, D. (2014), *Bayesian Data Analysis* (3rd ed.), Boca Raton, FL: CRC Press. [466]
- Gotwalt, C. M., Jones, B. A., and Steinberg, D. M. (2009), “Fast Computation of Designs Robust to Parameter Uncertainty for Nonlinear Settings,” *Technometrics*, 51, 88–95. [459,464,466]
- Hamada, M., Martz, H. F., Reese, C. S., and Wilson, A. G. (2001), “Finding Near-Optimal Bayesian Experimental Designs via Genetic Algorithms,” *The American Statistician*, 55, 175–181. [464]
- Huan, X., and Marzouk, Y. M. (2013), “Simulation-Based Optimal Bayesian Experimental Design for Nonlinear Systems,” *Journal of Computational Physics*, 232, 288–317. [469]
- Jones, M., Goldstein, M., Jonathan, P., and Randell, D. (2016), “Bayes Linear Analysis for Bayesian Optimal Experimental Design,” *Journal of Statistical Planning and Inference*, 171, 115–129. [458]
- Loeppky, J. L., Sacks, J., and Welch, W. J. (2009), “Choosing the Sample Size of a Computer Experiment: A Practical Guide,” *Technometrics*, 51, 366–376. [461]
- Meyer, R., and Nachtsheim, C. (1995), “The Coordinate Exchange Algorithm for Constructing Exact Optimal Experimental Designs,” *Technometrics*, 37, 60–69. [459]
- Mira, A., Solgi, R., and Imparato, D. (2013), “Zero Variance Markov Chain Monte Carlo for Bayesian Estimators,” *Statistics and Computing*, 23, 653–662. [469]
- Morris, M. D., and Mitchell, T. J. (1995), “Exploratory Designs for Computer Experiments,” *Journal of Statistical Planning and Inference*, 43, 381–402. [465]
- Müller, P. (1999), “Simulation-Based Optimal Design,” in *Bayesian Statistics* (Vol. 6), eds. J. M. Bernardo, M. J. Bayarri, J. O. Berger, A. P. Dawid, D. Heckerman, and A. F. M. Smith, Oxford, pp. 459–474. [459,464]
- Müller, P., and Parmigiani, G. (1996), “Optimal Design via Curve Fitting of Monte Carlo Experiments,” *Journal of the American Statistical Association*, 90, 1322–1330. [458,460]
- Müller, P., Sanso, B., and De Iorio, M. (2004), “Optimal Bayesian Design by Inhomogeneous Markov Chain Simulation,” *Journal of the American Statistical Association*, 99, 788–798. [459]
- Ntzoufras, I., Dellaportas, P., and Forster, J. J. (2003), “Bayesian Variable and Link Determination for Generalised Linear Models,” *Journal of Statistical Planning and Inference*, 111, 165–180. [467]
- O’Hagan, A., and Forster, J. J. (2004), *Kendall’s Advanced Theory of Statistics, Volume 2B: Bayesian Inference* (2nd ed.), London: Arnold. [467]
- Overstall, A. M., Woods, D. C., and Adamou, M. (2017), *Acebayes: Optimal Bayesian Experimental Design Using the ACE Algorithm*, R package version 1.3. [459]
- Pawitan, Y. (2001), *In All Likelihood: Statistical Modelling and Inference Using Likelihood*, Oxford: Oxford University Press. [461,466]
- Pichney, V., Ginsbourger, D., Richet, Y., and Caplin, G. (2013), “Quantile-Based Optimization of Noisy Computer Experiments With Tunable Precision” (with discussion), *Technometrics*, 55, 2–36. [469]



- Rasmussen, C. E., and Williams, C. K. I. (2006), *Gaussian Processes for Machine Learning*, Cambridge, MA: MIT Press. [460]
- Ripley, B. D. (1987), *Stochastic Simulation*, New York: Wiley. [469]
- Ryan, E. G., Drovandi, C. C., McGree, J. M., and Pettitt, A. N. (2016), “A Review of Modern Computational Algorithms for Bayesian Optimal Design,” *International Statistical Review*, 84, 128–154. [458,463]
- Ryan, E. G., Drovandi, C. C., Thompson, M. H., and Pettitt, A. N. (2014), “Towards Bayesian Experimental Design for Nonlinear Models That Require a Large Number of Sampling Times,” *Computational Statistics and Data Analysis*, 70, 45–60. [459,464]
- Ryan, K. J. (2003), “Estimating Expected Information Gains for Experimental Designs With Application to the Random Fatigue-Limit Model,” *Journal of Computational and Graphical Statistics*, 12, 585–603. [463]
- Santner, T. J., Williams, B. J., and Notz, W. I. (2003), *The Design and Analysis of Computer Experiments*, New York: Springer. [461]
- Severini, T. A. (2000), *Likelihood Methods in Statistics*, Oxford: Oxford University Press. [463]
- Waite, T. W., and Woods, D. C. (2015), “Designs for Generalized Linear Models With Random Block Effects via Information Matrix Approximations,” *Biometrika*, 102, 677–693. [466]
- Wang, L., and Zhang, L. (2006), “Stochastic Optimization Using Simulated Annealing With Hypothesis Test,” *Applied Mathematics and Computation*, 174, 1329–1342. [461]
- Weaver, B. P., Williams, B. J., Anderson-Cook, C. M., and Higdon, D. M. (2016), “Computational Enhancements to Bayesian Design of Experiments Using Gaussian Processes,” *Bayesian Analysis*, 11, 191–213. [458]
- Woods, D. C., Lewis, S. M., Eccleston, J. A., and Russell, K. G. (2006), “Designs for Generalized Linear Models With Several Variables and Model Uncertainty,” *Technometrics*, 48, 284–292. [464,467]
- Wu, C. F. J., and Hamada, M. (2009), *Experiments: Planning, Analysis and Optimization* (2nd ed.), Hoboken, NJ: Wiley. [459]